TDB-ACC-NO: NN9204459

DISCLOSURE TITLE: Customized Handling of Application Errors Under OS/2.

PUBLICATION-DATA: IBM Technical Disclosure Bulletin, April 1992, US

VOLUME NUMBER: 34

ISSUE NUMBER: 11

PAGE NUMBER: 459 - 461

PUBLICATION-DATE: April 1, 1992 (19920401)

CROSS REFERENCE: 0018-8689-34-11-459

DISCLOSURE TEXT:

- Disclosed is an OS/2* application called the Error Handler, which can generically handle errors which are generated by custom applications running in an OS/2 environment.

- Customarily when a OS/2 application is developed, the application developer is responsible for handling errors which could result from the application. The following is a brief example:

```
Begin
 open file
  if file not there
    then
     print an error message "Error File Not Found"
       End.
```

- With the Error Handler application, when a program detects an error it can send an error message to the error handler. The error handler application will then be responsible for processing the error.

- The following paragraph outlines the contents of an error message sent to the Error Handler. Each error is identified by a two-character prefix and a three-digit error number.

The error prefix
is used to type errors. For example, an error prefix 'CO', may be
selected to represent a communication error. An error number
identifies a specific error within a given error type. For example,
error CO 001 may be defined to be a communication error caused by a
bad token ring cable connection. The error prefixes and error
numbers are specified at system design time.
The error message
received by the error handler process contains the following
information:
-origin application name
-error prefix
-error number
-error arguments (up to 5)
The error handler application, upon retrieving an error message
from its interprocess message queue, determines the error number and
error prefix. Once the error prefix is determined, the error handler
constructs a filename by substituting the error prefix in to the
template 'ERRTXT__.MSG in place of the underline characters. For
example, when a CO type error is received, the file name
'ERRTXTCO.MSG' is constucted. This filename is then used as the name
of the error description filename. The error description file
contains descriptive text for each of the different errors of a given
type.
After the error description filename is built, the record
number that corresponds to the error number is read from this file.
- The record read from the description file has the following
format: ***** SEE ORIGINAL DOCUMENT *****
When the description is retrieved, the error handler builds an
error log record. The error log record contains the following
information:

- origin process name
- error prefix
- error number
- error description
- date and time of the error receipt

The error description text may contain place-holder symbols, such as %s, as part of the description. In such cases, the %s symbols are replaced by the error arguments which are received with the error message. This allows processes to pass text parameters to the error handler.

For example, if an error description text reads as 'Product with Serial No %s failed', and the originator of the error passes the '001897' as an error argument, the error handler will create the descriptive text: 'Product with Serial No 001897 failed'. Up to 5 substitutions can be made in each error description. The error arguments 1-5 are substituted from left to right in place of %s place-holder symbols.

- The Error Opcode is a code which tells the error handler application what actions to take when an error is detected. After the error opcode and the error description are retrieved, the error handler process executes the actions specified by the error opcode.

The possible actions that may be taken are:

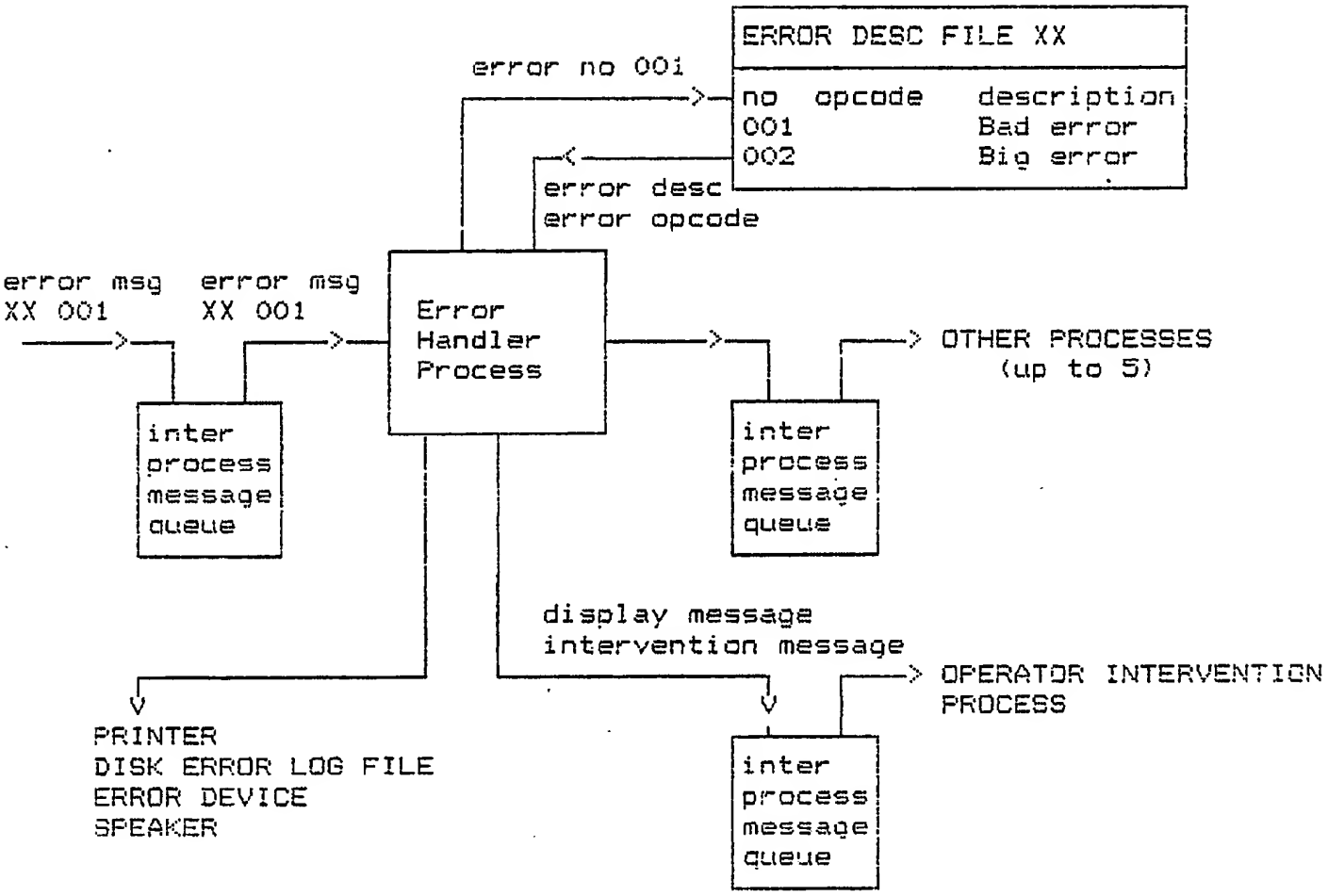
- -Logging the error to a disk file
- -Logging the error to a printer
- -Logging the error to a device
- -Forwarding the error to a specified operator interface for display
- -Beeping the PC speaker.
- -Forwarding the error to one or more operator intervention processes
- -Forwarding the error to other processes (up to 5).
- -Take no action - The error handler also allows additional opcodes to be defined (i.e., Log the error in the OS/2 Database)

The error handler process code does not require customizing.

After system requirements are determined and all system errors are identified, error opcodes and error descriptions can be created and stored in fixed record size ASCII text files for each error type (prefix).

- The figure (dataflow diagram) illustrates the typical operation of the Error Handler Application.

* Trademark of IBM Corp.


ERROR DESC FILE XX
no opcode description
001 Bad error
002 Big error
error no 001
error desc
error opcode
error msg
XX 001
error msg
XX 001
Error
Handler
Process
OTHER PROCESSES
(up to 5)
inter
process
message
queue
inter
process
message
queue
display message
intervention message
OPERATOR INTERVENTION
PROCESS
PRINTER
DISK ERROR LOG FILE
ERROR DEVICE
SPEAKER
inter
process
message
queue